This transaction is used to fill the customer tables of the Profile Generator the first time the Profile Generator is used, or to update the customer tables after an upgrade. The customer tables of the Profile Generator are used to add a copy of the SAP default values for the check indicators and field values. These check indicators and field values are maintained in transaction SU24. If you have made changes to check indicators, you can compare these to the new SAP default values and adjust your check indicators if necessary.

Step 1 – Initial Fill of the Customer Tables

Use this function for the initial transfer of SAP data (SU22) to the customer area (SU24). You only need to perform step in the development system. All subsequent systems in the landscape are updated by the transport created in step 3.

If authorization default values have already been maintained in this system, use step 2a to make changes to the default values from SU22.

If you want to use this function to reset the SU24 default values to the original SAP values, use the menu to navigate to Profile Generator Step 1.Caution: This action deletes all customer changes to default values. Use step 3 first, to be able to restore the initial situation, if necessary.

Steps 2a to 2d

If you have used the Profile Generator in an earlier Release and want to compare the data with the new SAP defaults after an upgrade, use steps 2a to 2d.

- Step 2a
  is used to prepare the comparison and must be executed before the other steps can be performed.

- Step 2b
  If you have made changes to check indicators or field values in transaction SU24, you can compare these with the new SAP default values. The values delivered by SAP are displayed next to the values you have chosen so that you can adjust them if necessary.
  If you double-click on the line, you can assign check indicators and field values. You maintain these as described in the documentation

for transaction SU24.

Note on the list of transactions to be checked

To the right of the list you can see the status which shows whether or not a transaction has already been checked. At first the status is set to to be checked.
If you choose the transaction in the change mode and then choose save, the status is automatically set to checked.
By choosing the relevant menu option in the list of transactions you can manually set the status to checked without changing check indicators or field values, or even reset this status to to be checked.
If you want to use the SAP default values for all the transactions that you have not yet checked manually, you can choose the menu option to copy the remaining SAP default values.

o Step 2c

This step shows all roles for which the menu applications are affected by changes to the authorization defaults. Roles for which the authorization data needs to be merged are set to the status "Profile comparison required" (merge mode) and have a red traffic light. If you then navigate to the authorization data maintenance tool (transaction PFCG), this leads to automatic merging.
If you do not want the system to automatically perform the merge, you can deactivate merge mode for selected roles using the corresponding function in the application toolbar. These roles are indicated in yellow. Merging is not possible in this status. You can navigate to the authorization data only in dispay mode.
Roles for which you have merged and saved the authorization data are marked as green and cannot be returned to merge mode. Selecting these roles has no effect.
Within a single call of this step, you can switch between active and inactive merge mode as often as you want. If you call the step again, the merge mode is automatically activated again, other than for roles with green traffic lights.
If a large number of roles (several thousand) need to be revised, to ensure smooth test operation, it can be useful to temporarily assign the subprofile of SAP_NEW associated with the new release (not the entire SAP_NEW profile!) to the users. This means that you then have the time to adjust the roles in stages. On the other hand, there is a certain security risk associated with this measure, since the new subprofile of SAP_NEW contains full authorization for all newly-delivered authorization objects.

Note also:

- The length of time taken to display the results list depends on the total number of roles that exist in the current client. If there are thousands of roles to be checked, expect a runtime of minutes.

- You can navigate from the results list to the authorization data either by single-clicking the desired role anem or by double-clicking any other places in the same line.

- Step 2c only takes account of roles in the current client. If you are operating role development in multiple clients, you need to perform this step in each of those clients.

o Step 2d

Transactions in the SAP system are occasionally deleted or replaced by other transactions.

In this step, you create a list of all roles that contain obsolete transactions in the role menu. This means the following:

· The transaction no longer exists.

· The application start lock has been set by SAP (transaction SM01_DEV).

· In S/4HANA: The transaction is obsolete and can no longer be used.

· The transaction is replaced by one or more other transactions.

You can use this program to delete or replace entries in the role menu.

Step 3

This step transports the changes made in steps 1, 2a, and 2b.

Adjusting the Authorization Checks

This points combined in this area are used to make changes to the authorization checks.

o Changes to the check indicators are made in step 4. You can also go to step 4 by calling transaction SU24.

- You can then change the authorization checks within a transaction.

- If the check indicator for an authorization object is set to

" Chec " with " Proposal:Yes ", the associated default values are k
included in the role when the authorization data is merged in transaction PFCG.

- If its check indicator is set to " Do Not Check ", the system does not check the authorization object in the relevant transaction.

- You can also edit authorization templates that can be added to the authorizations for a role in the Profile Generator. These are used to combine general authorizations that many users need. SAP provides a number of templates. You can either add these directly to roles, or copy them and edit them into your own templates. You can also add your own templates to roles.

- For information about the meaning of the check indicators, refer to the general documentation.

o In step 5, you can deactivate authorization object checks system-wide.

o In step 6, you can create roles from authorization profiles that you created manually.

Short text

Check transaction start authorization at CALL TRANSACTION

Title

Maintaining Transaction Start Authorization for CALL TRANSACTION

Use

In releases prior to release 4.6B, if a transaction was called from within another transaction using the ABAP command CALL TRANSACTION, the system did not check the transaction start authorization of the transaction that was called. As of release 4.6C, the developer of the calling transaction can define whether the transaction start authorization should be checked or not. This setting is made using program TASAUTH or transaction SE97. The developer's specification is transported as part of the calling transaction.

CAUTION

The statement CALL TRANSACTION without the addition WITH AUTHORITY-CHECK or WITHOUT AUTHORITY-CHECK is now **obsolete**. Statements with

these additions ignore configurations made in SE97. Statements without these additions function as described here.

Requirements

The program TASAUTH or transaction SE97 uses the TCDCOUPLES table. A trace function in the kernel enters the call couples in this table. Ideally, the specification whether the authorization is to be checked or not is maintained when the CALL TRANSACTION command is run for that call couple for the first time, since the call is only written in the table at this point. However, if this is not possible, the developer can also enter the call couples manually. He or she can also delete call couples.

Selection

The developer can select the calling transaction(s) in the selection screen. By selecting the corresponding radio buttons, he or she can select according to the transaction name or the development class and person responsible as defined in TADIR.

Output

If the developer has specified selection criteria for the calling transactions that are fulfilled by more than one transaction, the transactions are displayed in a selection list.

A green traffic light in the first column of this list means that either the system did not find any call relationships between the calling transactions and called transactions in the TCDCOUPLES table, or that for all the call relationships found it has already been specified whether the transaction start authorization is to be checked or not. A red traffic light means that there is at least one call relationship which has not been maintained.

Clicking on the name of the calling transaction displays a list of the called transactions.

If the developer only specified one calling transaction on the selection screen, the list of called transactions is displayed immediately.

Activities

In the list of called transactions, the developer should now specify for each called transaction whether the transaction start authorization of the called transaction is to be checked or not. To do this, he or she must click the corresponding column at the beginning of the row. The column headings have the following meanings:

- U = Not maintained
- N = Do not check

- P = Check

If the developer knows for certain that a called transaction is not in the list, he or she can enter the transaction manually using the 'Add Transaction' function.

Incorrect entries (transaction is not in fact called) can be deleted from the table using the 'Delete transaction' function.

The settings are saved in the TCDCOUPLES table by saving (for example, by choosing the save icon).

Process the dialog boxes of the transport system as you normally would.

Example

A developer wishes to specify for all transactions for which he or she is responsible whether the transaction start authorization is to be checked or not when transactions are called using the CALL TRANSACTION command. He or she has already tested all his transactions, therefore the trace function in the kernel has already entered the call couples in the TCDCOUPLES table.

The developer starts transaction SE97. On the selection screen, he or she selects the radio button 'Select calling transactions by development class/person responsible'. He or she leaves the 'Development class' field empty; his or her own user ID is already entered in the 'Person responsible' field. Choosing 'Execute' (F8) starts the selection of the calling transactions.

The selection list of the calling transactions is displayed. The developer sees several red traffic light symbols in the left column. He or she clicks the name of the first transaction with a red symbol. A list of the transactions called by this transaction is displayed.

Since this is the first time that the developer is maintaining the transactions, he or she has not yet specified for all transaction calls whether or not the transaction start authorization is to be checked. Therefore, a red cross meaning 'Not maintained' (column title 'U') is displayed in the left column of all the called transactions. A dot acts as a placeholder in the following two columns ('N' for 'Do not check' and 'P' for 'Check'). The developer clicks on one of the dots for each called transaction, depending on whether the transaction start authorization is to be checked or not. The red cross then disappears and a green tick appears in the specified column. Once the developer has specified whether or not the check is to take place for each called transaction, he or she saves his settings.

The developer goes back to the list of calling transactions. He or she consecutively selects each calling transaction with a red traffic light symbol and specifies in the list of called transactions whether the authorization is to be checked or not.

Definition

The authorization object is used to protect the roles. Roles are used to combine users in groups and to assign them different attributes, in particular transactions and authorization profiles.

Together with authorization objects **S_USER_GRP**, **S_USER_AUT**, **S_USER_PRO**, **S_USER_TCD**, and **S_USER_VAL**, you can use this authorization object to distribute user administration, if different administrators are to administer the users.

Defined fields

ACT_GOUP

- The field determines which roles may be processed.

ACTVT

- The field determines which activities may be executed using the roles. You can choose from the following activities:
    - **01**
      Create roles

- **02**
  Change roles
- **03**
  Display roles
- **06**
  Delete roles
- **08**
  Display change documents for roles
- **22**
  Compare role user master records
  Roles are assigned to users with this.
- **36**
  This activity is not yet used. It is planned for use for additional objects that can be maintained from the roles.
- **21**
  Transport role
- **59**
  Distribute roles to another system using RFC
- **64**
  Generate authorization profiles from role.
- **68**
  Modeling: Assigning roles to systems or users in user management using models. The actual assignments can be derived from these models later.
- **78**
  Assign roles to systems or user groups in central system of central user administration
- **79**
  Assign single roles to composite roles
- **DL**
  Download
  Save roles to a file
- **UL**
  Upload
  Upload roles from a file

User Master Maintenance: User Groups

Definition

Authorization object that is checked during user administration (*Tools -> Administration -> User Maintenance*) in the following transactions:

- SU01 (Maintain Users)
- SU10 (Delete or add a profile for all users)
- SU12 (Delete all users)

The check is made in the following role maintenance transactions:

- PFCG (assign users to roles)
- SUUM, SUUMD: User administration in the central system of Central User Administration

Defined fields

The object is defined with the following two fields:

- *User Group*: Can be used to define the user administrator for the maintenance of one or more user groups. This makes sense if there is to be more than one user administrator.

  Every user must then be assigned to a user group. Users that are not assigned to a group can be maintained by all user administrators.
  As of release 4.6, users can be assigned to more than one group. Note that one group is marked as relevant for the authorization check. This is the group listed under the 'Logon Data' tab in the user maintenance transaction SU01.

- *Activity*: This field can be used to limit what the administrator is allowed to do with the authorization.

  Possible values:

  - **01**: Create
  - **02**: Change
  - **03**: Display

- **05**: Lock, unlock
- **06**: Delete
- **08**: Display change documents
- **22**: Add users to roles
- **24**: Archive
- **36**: Extended maintenance
- **50**: Move
- **78**: Assign
- **68**: Model: Model the assignment of systems or roles to users in user administrations, so that you can later derive the actual assignments from this model.
- **PP**: Set productive password

Example

Field	Values
User Group	SUPER
Activity	01 - 03, 06

This authorization allows the user administrator to create, maintain, delete, and display users in the SUPER group or users not assigned to a group. However, the administrator may not lock or unlock the users or display change documents.

Further notes

- Copy clients: An authorization for this object is only required if you want to copy user masters and authorizations into the new client prior to the copy process. More information is available in the SAP Library: SAP NetWeaver Library -> SAP NetWeaver Library: Function-Oriented View -> Application Server -> Application Server ABAP -> Administration of Application Server ABAP -> Change and Transport System -> BC - Client Copy and Transport (BC-CTS-CCO).

User Master Maintenance: Authorizations

Definition

Authorization object, which is checked during authorization maintenance.

The check is made in the following authorization maintenance transactions (*Tools -> Administration -> User maintenance*).

- SU03 - maintenance of authorizations
- SU02 - allocate authorization to a profile

Defined fields

This object is defined with the following three fields:

- *Authorization object*: The authorization objects, for which the authorization administrator may maintain, delete, etc., authorizations, must be entered here.
- *Authorization name*: With this field, the authorization administrator can be given a limited name space for the assignment of authorization names.

  It makes sense to do this when there are several adminstrators, so that the authorizations are not maintained reciprocally or possible internal naming conventions are adhered to.
- *Activity*: This field can be used to limit what the administrator may do with the authorization.

  Possible values:

  - 01 = create
  - 02 = change
  - 03 = display
  - 06 = delete
  - 07 = activate
  - 08 = display change documents
  - 22 = assign authorization profiles
  - 24 = archive

Example

Field	Values
Authorization object	S*
Authorization name	*
Activity	01 - 06

With this authorization, the authorization administrator can create, change, delete, and display authorizations of any name for all objects beginning with S (Basis authorization objects). The administrator can not, however, activate the authorizations or display the change documents.

User Master Maintenance: Authorization Profile

Definition

Authorization object, which is checked during authorization maintenance.

The check is made in the following profile maintenance transactions (*Tools -> Administration -> User maintenance*).

- SU02 (Maintain profiles)
- SU01 (Assign profile to user)
- SU10 (Assign profile to all users or remove assignment to all users)

Defined fields

The object is defined with the following fields:

- *Authorization profile*: In this field, you must enter the authorization profiles that an authorization administrator may maintain or that a user administrator may assign to his users.
- *Activity*: In this field, you can limit what the administrator is allowed to do with the profile.

  Possible values:

  - **01**: Create
  - **02**: Change

- **03**: Display
- **06**: Delete
- **07**: Activate
- **08**: Display change documents
- **22**: Assign profile to users / remove assignment
- **24**: Archive

Example

Field	Values
Profile name	*
Activity	01 - 06

This authorization allows the authorization administrator to create, maintain, delete and display profiles at will. However, he may not activate profiles or display change documents.

Authorizations: Transactions in Roles

Definition

Authorization objects control the transactions that system administrators can assign to a role, as well as the transactions for which they can assign transaction code authorization (object S_TCODE).
Note that in the Profile Generator, you can only maintain intervals of transactions if you have full authorization for S_USER_TCD for authorization object S_TCODE. Otherwise you can only maintain individual values for the object S_TCODE.

Defined fields

TCD

- Transactions that administrators may assign to roles and for which they may assign authorization to start a transaction in the Profile Generator.

Authorizations: Field Values in Roles

Field value authorization for activity groups

This authorization object allows you to restrict the values an administrator is allowed to add or change for a role in the Profile Generator.

The authorization object refers to all field values except the values of the object S_TCODE.

- The authorization to add transactions to a role or change the transaction start authorization in a role is granted using the authorization object **S_USER_TCD**.

Defined fields

OBJECT

- Specification of the authorization object

- **Note**
  If a user wants to maintain organizational levels in a role, the user must have the complete authorization ("*") for this authorization field.

AUTH_FIELD

- Specification of the authorization field to be protected

AUTH_VALUE

- The permitted values for the field in this object. These are the values the user is allowed to add or change in the role.

- **Note**
  If a user is to be allowed to maintain intervals or use generic entries (a value with "*" at the end), the user must have the full authorization ("*") for this authorization field.